

## ACTIVIDADES PRELIMINARES

a. El ciclo de vida de la construcción de los programas es:

- Planificación y análisis de requisitos
- Diseño de la arquitectura del software
- Se escribe el código del programa siguiendo el diseño
- Se realizan pruebas para identificar errores y asegurando de que el programa funciona según lo esperado
- Se instala en el entorno de producción
- Después de implementarlo, se lleva a cabo el mantenimiento que incluye corrección de errores, mejoras y adaptaciones.

b. El análisis de problema es una etapa para el desarrollo de soluciones:

- Definición clara del problema: Esto implica especificar los síntomas, las causas y los límites del problema.
- Determinar qué resultados se esperan del proceso de resolución. Los objetivos deben ser claros, medibles y alcanzables.
- Recopilación de información
- Evaluación de limitaciones y restricciones.

c. El proceso de solución de problemas es:

- Identificar los problemas
- Analizar el problema
- Generar alternativas soluciones
- Seleccionar la mejor solución
- Implementar la solución
- Evaluar los resultados

D. Elementos que se deben entregar al cliente son:

- Documentación del software
- Código fuente y ejecutable



e

CLIENTE	El cliente es el banco que desea desarrollar el programa para el manejo de los cajeros automáticos
USUARIO	Son las personas que utilizan el cajero automático
REQUERIMIENTO FUNCIONALES	Retirar dinero, consulta de saldo, autenticación.
REQUERIMIENTO NO FUNCIONAL	Seguridad, Disponibilidad, tiempo de respuesta, usabilidad, mantenimiento.
MUNDO DEL PROBLEMA	Cajeros automáticos, Infraestructura bancaria, usuarios y su interacción, seguridad.

f

REQUERIMIENTO FUNCIONAL 1	Nombre	Registro de Usuario
	Resumen	Permitir a un nuevo usuario crear una cuenta en el simulador bancario ingresando sus datos personales
	Entradas	Nombre, apellido, correo electrónico, número de identificación
	Resultados	Se obtiene una cuenta creada con credenciales de acceso y saldo inicial definido



## REQUERIMIENTO FUNCIONAL 2

Nombre	Calcular el perímetro del triángulo
Resumen	El programa debe calcular el perímetro sumando las longitudes de sus tres lados
Entradas	Tres valores numéricos que representan los lados del triángulo
Resultados	El sistema muestra el valor del perímetro del triángulo

## REQUERIMIENTO FUNCIONAL 3

Nombre	Calcular el área del triángulo
Resumen	El programa debe calcular el área del triángulo
Entradas	Tres valores numéricos
Resultados	Mostrara el valor del área del triángulo

H

Entidad	Nombre	Descripción
Entidad	Triángulo	Representa la figura geométrica principal
Entidad	Punto	Representa un punto en el plano cartesiano, usando los vértices
Entidad	Operación geométrica	Representa cálculos que se pueden realizar

## • Puntos de Reflexión

1. ¿Qué pasa, si no identificamos bien las entidades del mundo?

• Podríamos definir atributos dentro de una entidad equivocada, lo que dificulta la organización del código.

2. ¿Cómo decidir si se trata de una identidad y no solo de una característica?

• Una identidad debe representar un concepto independiente y poder tener múltiples atributos o comportamientos.

i

Atributo	Valores posibles	Diagrama UML
Número cuenta	Número único de la cuenta	<pre> classDiagram     class CuentaBancaria {         NumeroCuenta         Titular         Saldo     }         </pre>
Titular	Nombre del propietario	
Saldo	Monto disponible en la cuenta	

Atributo	Valores posibles	Diagrama UML
Límite Sobregiro	Cantidad máxima que se puede girar	<pre> classDiagram     class CuentaCorriente {         LimiteSobregiro         CargoMensual         Tipo     }         </pre>
Cargo Mensual	Comisión mensual por el uso de la cuenta	
Tipo	Indica si es personal o empresarial	



Atributos	Valores Posibles	Diagrama UML
Tasa Interés	Porcentajes de interés aplicado a la cuenta	<pre> classDiagram     class CuentaAhorros {         Tasa Interés         Saldo mínimo     } </pre>
Saldo Mínimo	Monto mínimo requerido para evitar penalizaciones	

Atributos	Valores Posibles	Diagrama UML
Nombre Mes	Nombre del mes	<pre> classDiagram     class Mes {         Nombre Mes         Dias     } </pre>
Cantidad de días del mes	Días de mes	

## Reflexión sobre el nivel de precisión en un algoritmo

### 1. Falta de información clave:

- Si el algoritmo no especifica detalles como dirección o estación, los usuarios se pueden perder.

### 2. Ambigüedades en las instrucciones:

- Expresiones como "tome el metro" pueden ser interpretadas de distintas maneras.
- Indicaciones vagas pueden ser confusas.

## Reflexión sobre dibujar/escribir un algoritmo

1. **Dibujo:** En el primer cuadrante, se realiza un dibujo simple.

2. **Algoritmo:** En el segundo cuadrante, se escriben instrucciones detalladas para replicar el dibujo.

3. **Ejecución del algoritmo.**

## Síntesis de los resultados obtenidos.

- Dificultad en la interpretación de instrucciones
- Importancia de la precisión y el detalle
- Limitaciones del lenguaje natural
- Relación con la Programación



## Ejemplo N° 2: Elecciones

K. Redacta el enunciado del problema y el nombre, cada uno de los requerimientos funcionales.

### Solución

Se requiere diseñar y desarrollar una aplicación informática que permita gestionar la información y estadísticas de unas elecciones con tres candidatos. La aplicación debe ser capaz de:

- Visualizar información detallada de cada candidato
- Registrar votos y actualizar la información correspondiente
- Calcular estadísticas generales y por rango de edad y género
- Proporcionar información precisa y actualizada sobre las elecciones.

Además, la aplicación debe considerar el impacto de la publicidad en la campaña electoral, incrementando el costo de la campaña según el medio de comunicación utilizado.

### Requisitos

- La aplicación debe ser fácil de usar y proporcionar información precisa y actualizada
- Debe ser capaz de gestionar información de tres candidatos

- Debe considerar el impacto de la publicidad en la campaña electoral.

### "Requerimientos funcionales"

R<sub>1</sub> - Visualizar la información de un candidato.

- \* Permite visualizar la información detallada (Nombre, edad, Partido político, costo campaña, Cantidad votos, y % de voto.) de cada candidato
- 

R<sub>2</sub> - Votar por un Candidato

- \* Permite votar por un candidato
- 

R<sub>3</sub> - Calcular el número total de votos

- \* Permite conocer la cantidad total de votos que hay en la urna, es decir la suma de los votos que hay por cada uno de los candidatos
- 

R<sub>4</sub> - Calcular el número total de votos.

- \* Permite conocer la cantidad total de votos que hay en la urna, es decir la suma de los votos que hay por cada uno de los candidatos
- 

R<sub>4</sub> - Calcular el costo promedio por campaña

- \* Permite conocer el promedio del costo de campaña de los tres candidatos, es decir, la razón de la sumatoria de los costos



de la campaña de los tres condados y el número total de candidatos.

---

R<sub>5</sub> - Visualizar estadísticas

\* Permite visualizar estadísticas de votos por género (femenino y masculino) y su rango de edad (joven, media mayor)

---

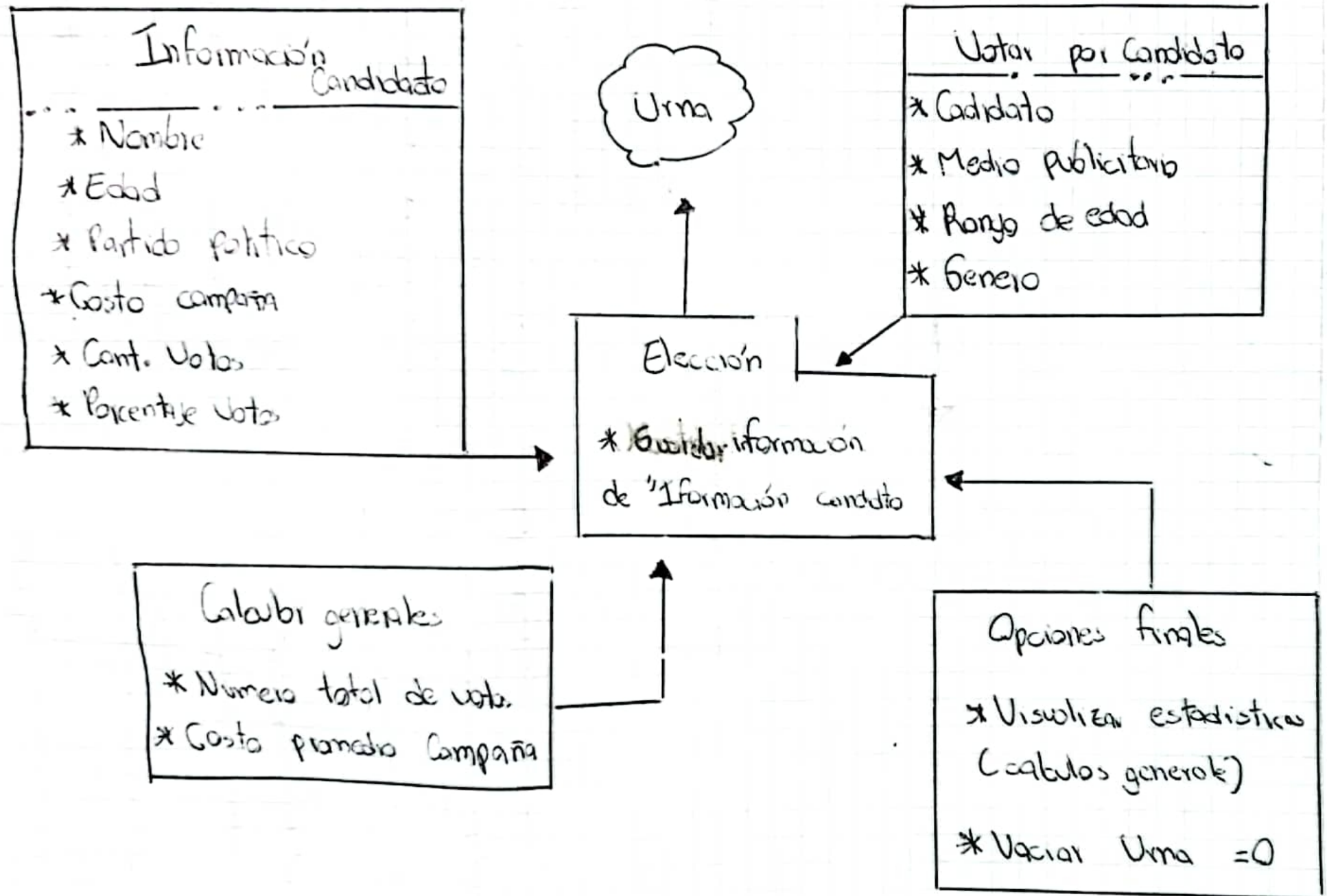
R<sub>6</sub> - Vaciar la urna

\* Permite reiniciar los votos y los costos de campaña de todos los candidatos, de tal manera que estos queden en cero.

1. Dibujar

Diagrama de casos

# Diagrama de Uso





m. Escribe el nombre de cada una de las clases identificando sus respectivas variables (atributos) y funciones:

"Mundo"

Variables: \* Urna

- Candidatos
- Calcular votos fem/mas
- Registrar voto
- Calcular votos (edad)
- Costos Promedio (campana)

\* Candidato

- Costos (Inter, Radio, TV)
- Información candidato  
(Nombre, apellido, partido poli, edad, Costo campana, Int)

\* Votos

- Cantidad (Mascul/femen)edad.

# "Interfaz"

Variables: \* Candidato

- Información Candidato
- Voto por edad

\* Elecciones

- Registro Voto
- Porcentaje Voto

\* Estadísticas Candidato

- (Candidato)

\* Votar

- Número candidato
- Género, Edad y influencia.

\* Opciones

- ESTADÍSTICAS
- VACIAR URNA

\* Estadísticas Generales

- Votos
- Costo Promedio Campaña
- Género, Edad,



## "Pruebas"

### \* Urna test

- Ingresar voto
- Registrar voto
- Calcular Cuentas
- Calcular (Ganador, edad, etc.)

### \* Candidato

- Candidato
- Registrar voto
- Reiniciar

### \* Votos Rango

- Votar Rango (edad)
- Registro
- Reiniciar